Visualization in Scientific Computing – Where do we go from here?

Arthur J. Olson, Ph.D. The Scripps Research Institute

In the years leading up to the first ViSC Report in 1987, computer graphics development was principally driven by scientific and technical applications. In the intervening years scientific visualization has been propelled forward by the hardware advances that have continuously tracked Moore's Law and the commoditization of 3D graphics accelerators. Computer graphics software development has been principally driven by the requirements of the entertainment industry for high quality rendering in films and interaction environments of video games. An analysis of commercial scientific visualization software over the past decade shows surprisingly little growth.

The questions we should now ask are: 1) where are the new frontiers of visualization and how will the unmet needs of scientists and engineers drive future development? 2) While graphics hardware is rapidly approaching the resolution and bandwidth parameters to match human visual capacity, how can improved human-computer interface approaches address the more difficult task of assisting the processes of analysis, discovery and knowledge transmission? 3) How can we aid in the development of useful and useable tools for specific scientific applications?

Answers to these questions revolve around developments in several related technologies:

1) Development of abstract information visualization and representations for high dimensional data and error analysis.
2) Expansion of the human computer interface beyond the purely "visual" paradigm to enable "direct manipulation" and "natural interaction" with data and interactive navigation/simulation/computation.
3) Creation of interaction environments to enable custom applications and interfaces via end-user development.

We should be looking toward a future of rapidly evolving computational technology: where computing is embedded in almost everything and connected wirelessly everywhere; where custom physical devices can be generated as computer output; and where algorithms and software components can be integrated and evolve to produce novel applications built by non-programming domain experts and end users. We should also be looking at the comparatively static nature (slowly evolving) of the human condition by understanding and integrating our growing knowledge of perception and behavior into our computer interfaces.